

A New Paradigm for Optimizing Hybrid Simulations of Rare Event Modeling for Complex Systems

Paola Bandini, Department of Civil Engineering, *paola@nmsu.edu*
Jeanine Cook, Department of Electrical and Computer Engineering, *jcook@nmsu.edu*
Martha C. Mitchell, Department of Chemical Engineering, *martmitc@nmsu.edu*
Linda Ann Riley, Department of Industrial Engineering, *linriley@nmsu.edu*
New Mexico State University
P.O. Box 30001, MSC 4230
Las Cruces, New Mexico 88003

Keywords: Hybrid simulation, combined simulation, rare event modeling, engineering modeling, complex systems

ABSTRACT

This article proposes a framework for advancing and disseminating new, fundamental knowledge in modeling and simulating rare events in large-scale complex systems. Most complex systems such as our cities, national air space, nuclear power plants, corporate enterprises, and global logistics systems are characterized by both embedded processes exhibiting continuous state changes, and stochastically and/or deterministically occurring discrete events. (Systems possessing these characteristics are typically termed hybrid and/or combined systems.) Traditionally, these systems have been modeled and simulated using either purely phenomenological models or discrete-event models because of the nature of the mathematical or physics equations underlying the models.

This article discusses the current practice of hybrid/combined system modeling and why this type of modeling will play a crucial role in the future when planning responses to rare events occurring in complex systems. It also proposes a framework supported by a pilot model for advancing knowledge in this field.

INTRODUCTION

Most complex systems such as our cities, national air space, nuclear power plants, theme parks and global logistics systems fall in the category of hybrid and/or combined systems, which are characterized by both embedded processes exhibiting continuous state changes and stochastically and/or deterministically occurring

discrete events. Traditionally, these systems have been modeled and simulated using either purely phenomenological models or discrete-event models because of the nature of the mathematical or physics equations underlying the models.

However, the full development of realistic computer modeling that allows for the optimization and scenario analysis of these large-scale hybrid environments has not evolved to its fullest extent for several reasons. These reasons include:

1. differences in mathematical and modeling approaches in various engineering disciplines;
2. lack of an agreed upon all-purpose modeling code or software;
3. slow progress in defining and creating workable interfaces for combined, hybrid modeling that incorporates the “best” modeling and simulation codes from each engineering discipline;
4. information technology barriers involving computational speed and algorithmic efficiency that inhibit real-time responses to rare or catastrophic events;
5. methodological differences in optimization approaches and algorithms for continuous versus stochastically defined discrete systems; and
6. data gathering and processing challenges associated with modeling real-time complex systems in both continuous and discrete-event environments.

Modeling and simulation is a central facet to all engineering disciplines. However, the exact nature of modeling and simulation, the tools and code used, the conceptualization of the problem, the scope of the problem under analysis and the computational resources required to provide useful results for each discipline differs greatly. Although much of the focus in the engineering classroom is on discipline-specific approaches for modeling and simulating either continuous

or discrete-event based systems, there has been significant research over the past 35 years in attempting to articulate formalisms for hybrid simulation.

Fahrland [1970] is generally credited with the initial work on modeling hybrid systems. Even in his earliest work, Fahrland outlined many of the limitations and barriers associated with modeling the integration of continuous and discrete state systems. He noted that these hybrid models require overly restrictive definitions of scope, introduce software interface complexities, and lead to computational inefficiency.

Since Fahrland, other researchers have proposed different approaches to modeling combined continuous/discrete systems that attempt to overcome known IT challenges to accomplishing this task. Some examples of approaches include hybrid agent-based simulation [Lee 2001], hybrid automata [Alur, 1995; Herzinger, 1996], abstract dynamical modeling [Branicky, 1996], high level architecture developed in 1996 by the Department of Defense, and dynamic structure discrete event system modeling [Barros, 1996].

When modeling these hybrid systems, the fundamental challenge to integration involves the state transition conceptualization. In discrete-event models, a constant time step is defined prior to simulation. Even though these time steps may be exceedingly small and thousands of events may occur at a single time step, the next state transition is driven by the events triggered by the present state and any new inputs to the system. Many time steps may occur with no state changes before the system continues its evolutionary process.

In continuous models, differential equations specify rates of change. Future states are computed through a time marching algorithm, (stepping in time), where next steps are predicted by extrapolating either difference equations or integral equations. As reviewed by Zeigler et al. [2000], the major drawback of simulating continuous models on digital computers involves the time step gap between two computation instances. Although the model in reality is operating continuously, calculations on the computer are undertaken without computation of the interim inputs and outputs. Further complications arise when simulating non-linear dynamical systems. Numerical integration methods are the generally accepted approach to simulating continuous models but introduce several additional challenges such as step approximation error, step dependence error and the start-up problem.

In all of the various approaches to hybrid modeling there is usually one environment, either discrete or continuous, that dictates the model development and consequently, the simulation development. For example, HLA, the Department of Defense required standard, combines parallel and distributed event modeling as its

infrastructure. Embedded, continuous systems characterized by ordinary differential equations (ODEs), partial differential equations (PDEs), differential algebraic equations (DAEs) and multi-domain PDEs are simulated using time-stepped numerical integration. Indeed, most hybrid approaches with a lead discrete-event paradigm use time-stepped numerical integration to simulate continuous subsystems. This is of course in contrast to the manner in which these subsystems are modeled, e.g. using primarily ODEs and PDEs.

As the level of complexity of hybrid systems grows, especially when characterized by random, stochastic events, so does the ability of the system to respond with more accurate output for decision makers. However as the hybrid system grows more complex, the time lag in providing real-time output to aid decision makers also grows. This is in part caused by the information technology constraints involving computational speed and efficiency demanded by the various statistical formulae governing statistical significance of the simulation results.

In addition, another impediment to real-time response is the varying requirements for continuous portions of the models with respect to computing resources and time required for numerical integration of complex multidimensional subsystems. Furthermore, as the optimization inputs grow, a second level of complexity is introduced by the number of time sampled runs and replications required for accurate analysis of the simulation results. Intelligent computational algorithms are needed to search the multi-dimensional feasible solution space of the simulation results. However, because of the complexity associated with the design and implementation of these algorithms, only simple hybrid systems have been successfully modeled.

In today's information technology-driven environment, the need for modeling hybrid, complex, large-scale systems is critical. Responding to natural disasters, evaluating terrorist attack scenarios, modeling breaches in national transportation networks, or analyzing impacts of rare events on the battlefield is essential to maintaining security of all sectors of society. Our research question allows us to explore a new paradigm that will emphasize the specialization and integration of modeling and simulation tools, and codes from each of our engineering disciplines. Unlike any other studies found in the literature, this research will focus on the required interfaces to create the new framework and not on the creation of an entirely new language or specification to model hybrid systems.

WORK TO DATE

In recent years, there has been limited published research on building hybrid/combined simulation models

that interface codes from both continuous and discrete-event languages. Law and Kelton [2000] suggest SLAM II[®], SIMAN[®], SIMSCRIPT[®] as primary codes allowing for the establishment of interfaces between different discipline-specific simulation tools. Klingener [1995] has defined an interface module for ProModel[®] software allowing for the integration of discrete events and continuous chemical industry processes. Music and Matko [1999] have proposed a combined simulation approach using Matlab[®] interfaced with Simulink[®] for control systems, while Sturgul and Prisbrey [2001] use Matlab[®] and the GPSS[®] simulation language for simulating a mineral processing crusher circuit. Some simulation software, e.g., GPSS[®] and ProModel[®] are capable of embedding, which consists of calling subroutines written in a host language, FORTRAN or C⁺⁺, for example, to substitute blocks and control statements [Schriber, 1986].

In addition to the efforts to interface continuous and discrete events using existing tools and languages, there is a large body of work that is directed toward developing new modeling languages and environments that specifically include combined modeling capabilities. Most of this effort comes from the European research community. Some of these modeling languages include Modelica, SMILE, Dymola, Allan, NMF, ObjectMath, SIDOPS+, and Omala.

Modelica [Mattsson et al, 1999] is a modeling environment that combines several of the features from various modeling languages into a uniform, object-oriented language for the modeling of physical systems. Modelica hybrid models are based on the synchronous data flow principle, which states that at every time instant, the active equations (either discrete or continuous) express relations between variables that must be fulfilled concurrently. Therefore, a hybrid model is expressed as a set of synchronous differential, algebraic and discrete equations that behave deterministically. Although Modelica enables interfacing of continuous and discrete event models, the modeler is relegated to model within the confines of the Modelica language and environment.

The language χ [Fabian, 1999] and its associated environment support the modeling and simulation of combined models. It is a concurrent language with the continuous time component based on differential algebraic equations and the discrete-event part based on communicating sequential processes (CSP [Hoare, 1978]). However, similar to Modelica, the modeler is restricted to defining a model within the χ environment.

Our approach will be different from the existing work in the field in that (1) the scope of the system will be greatly expanded, (2) the interface will be unique, (3) the interfaced simulation code will be drawn from different engineering disciplines, (4) system optimization and

scenario development will be introduced into the new paradigm, and (5) the paradigm will be developed for the purpose of modeling rare events.

The primary research focus of this project is to develop a platform to support the integration of different types of computational models into a single modeling environment. We plan to integrate execution-driven system models with physical models of bridge loading and airborne transport in the context of a bridge collapse triggering release of toxic contaminants. The integration primarily involves interfacing the various models to communicate with one another in a cooperative manner such that an event spawned in the master model, (the execution-driven simulation), is communicated in a context understood by the structural and molecular models. A means to transform the outputs from an execution-driven model into inputs to the physical models will be developed. Conversely, a means of transforming output from the physical models to input to the execution-driven model will also be implemented. We also plan to develop a visual interface between model types to provide the user with feedback during simulation of physical models that traditionally provide text-based output. Finally, we will develop a framework for distributing various model types (e.g., initially, structural and molecular only) across a network of computing resources.

This project will uniquely attempt to incorporate/interface various continuous and discrete models, specific to different engineering disciplines, into a master model without having to define the processes of the other (slave) models in the same programming language as the master model. Accomplishing this goal will allow for the implementation/interaction of more sophisticated, already-existing software from diverse disciplines for the solution of complex, real rare event systems.

Modeling and Simulation in Chemical Engineering

Modeling and simulation studies of chemical engineering processes run the gamut of macroscale to the microscale. On the macroscale simulations are used for plant design flowsheeting. Common software packages in use in academia and industry are Aspen PLUS[®], HYSYS[®] and ChemCAD[®]. These packages allow input of parameters and choice of thermodynamic modeling equations. They solve the mass and energy balance equations, either at equilibrium or using special continuous transient modules. Examples in the literature of studies using such software include work by [Lisitsyn et al., 2002] and [Smejkal and Soos, 2002]. Chemical engineers also develop continuous time models of chemical processes, including the differential equations arising from mass and energy balances on process units.

Particular emphasis is placed on the systems of differential equations that arise in chemical reactor design. In addition, thermodynamic properties of systems can be obtained from a macroscale modeling approach [Mitchell et al., 2002a; Mitchell et al., 2002b].

On the microscale, computational fluid dynamics is used to model flow patterns and temperature profiles in pipes, in reactors and during crystal growth. These models contain partial differential equations, which can be solved using finite element, finite volume or finite difference techniques. Examples in the literature include Lei et al. [2002], Lipchin and Brown [2000], and Vartak et al., [2000].

On the nanoscale, information can be obtained from studying the classical behavior of molecules in a system using Monte Carlo and molecular dynamics simulations, as well as Molecular Density Functional Theory models. These techniques use the principles of statistical mechanics. In addition, information about reactions can be obtained from *ab initio* methods.

Modeling and Simulation in Industrial Engineering

Modeling and simulation of complex systems in Industrial Engineering primarily focuses on the design of discrete-event simulation environments characterized by stochastic and probabilistic events evolving over relatively long periods of time. These computer-modeled environments are developed, run, and optimized either from a distributed platform or from a single location. Industrial engineers model all types of discrete-event systems, from health care applications to battlefield scenarios to manufacturing industries to people flow challenges. The scope of study for industrial engineers usually focuses on large systems...a city, airport, transportation network, retail complex, or hospital.

A number of different simulation codes and tools are available to industrial engineers. Although many industrial engineers program in C or C++, there are a number of off-the-shelf simulation tools that prove invaluable when modeling complex systems. Some of these include: Arena[®], Witness[®], Extend[®], ProModel Corporation's products ProModel[®], ServiceModel[®], MedModel[®], AutoMod[®], Optquest[®], and Matlab[®]. All of the aforementioned products include some type of optimization module as well as interfacing capabilities to object-oriented codes.

A current grand challenge in developing discrete-event simulation models is gaining statistically valid estimates of the true performance of a system, and then optimizing the almost infinite feasible solution space of the system. This proves to be an extremely computationally intensive problem. The most widely used search procedures for optimizing a discrete-event

simulation's feasible region are: deterministic search methods, pattern search methods, probabilistic search methods, and hybrid techniques. A great deal of the present work in optimization of discrete-event simulation is based on the probabilistic search techniques of simulated annealing [Liu, 1999; Zolfaghari and Liang, 1998] and genetic algorithms [Azadivar et al., 1999; Pierreval and Tautou, 1997]. A second area of algorithmic optimization development is in hybrid techniques primarily based on evolutionary strategies [Shi et al., 1999; Feyzbakhsh and Matsui, 1999].

New approaches to optimizing classical engineering problems are proposed by [Riley, 2000a; Riley, 2000b]. Here a great deal of work is concentrated on intelligent computing applications such as genetic algorithms, neural networks, and evolutionary programming [Riley and Flores-Mendoza, 2002; Riley and Salais, 2002]. Some of the application areas where intelligent computation development is focused involve large-scale facility design [Riley and Flores-Mendoza, 2001; Riley 1999], stochastic shortest path problems [Riley and Gomez-Sanchez, 2001], network modeling, intelligent agent modeling, and neuro-fuzzy scheduling. In all of these cases, fast and powerful computers are needed to efficiently handle the partitioning and management of data to address the unique searching requirements of the various optimization approaches.

Modeling in Civil Engineering

Most studies in Civil Engineering involve, but are not limited to, the stability analysis of structures, bearing capacity, soil-structure interaction, performance of construction materials, and fluid transfer in soils. Thanks to the latest significant advances in computer capabilities and the urgent need for design methods applicable to more realistic, complex systems, a considerable amount of research effort has been devoted to the development of new simulation tools that find specific applications in Civil Engineering. Optimization of the design methods in this area of study should be undertaken for both safety and cost reduction purposes.

Numerical simulations using the finite element method have been effectively applied to fluid flow models in soil [Choo and Yanful, 2000; Khire 2000], with special applications in pollutant transport and treatment and management of contaminated soils [Blunden and Indraratna, 2001], urban and air traffic flow, construction materials performance [Cervenka and Cervenka, 1996], and discrete event simulation in construction management [Luna and Wu, 2000] to mention a few. In Geotechnical Engineering, computer modeling has been shown to provide a more detailed response than laboratory test results, usually at a lower cost and with more versatility. In this field, finite element and numerical limit analysis

have been successfully used to analyze the stability of soil slopes under static [Yu et al., 1998; Kim et al., 1999, 2002] and dynamic loading conditions [Loukidis et al., 2003], as well as of shallow and deep foundations [Hu and Randolph, 1998; Hu et al., 1999; Bandini, 2002].

Numerical simulation of large-scale rare events can be used for attaining a better understanding of the part, or parts, of the model that are more susceptible to failure. This is especially significant for critical civil structures in which human lives and national safety could be in jeopardy – hospital, bridges, subways, fire stations – under terrorist attacks with large impact loads or due to catastrophic accidents. A research team led by M. Sozen [Sozen et al., 2002] at Purdue University has recently completed phase I of a computer simulation of the September 11 Pentagon attack using LS-Dyna[®] with the purpose of developing the appropriate tools for a detailed analysis of potential disasters before they occur. This is an example of the use of commercially available software (LS-Dyna[®] is normally used by auto-manufacturers to simulate car crashes) combined with technical and scientific expertise for practical applications different from those for which the computer package was originally developed.

Modeling in Electrical and Computer Engineering: Computer Architecture

Modeling and simulation in Electrical and Computer Engineering is performed in a broad range of discipline-specific areas including power, networking, and computer architecture. We believe power and networking to be the most relevant areas to the modeling and simulation of rare events in complex systems.

The area of computer architecture performance analysis and design is focused primarily on the modeling and discrete-event simulation of current and future microprocessors to analyze performance and/or make design decisions prior to implementation on chip. Modeling is performed on many different levels of granularity ranging from the transistor level to the functional level. Hybrid models typically interface very detailed models (cycle-accurate) with analytical and functional models [Cook, 2002; Loh, 2001]. As in other engineering areas that utilize modeling and simulation for development and analysis, the most detailed (microprocessor), models are the most accurate and the most time-intensive to simulate.

Large commercial companies develop performance models and simulators that are specific to microprocessor families that they produce. Because of the manpower, time, and monetary resources that are dedicated to this effort, these simulators are proprietary. Small commercial industries and academia must either develop their own models and simulators (which requires several

years with multiple developers), or they rely on simulators that have been developed by other academic institutions. SimpleScalar[®] [Berger, 1997], an academically developed, cycle-accurate microarchitecture simulator, is widely used by small businesses and academia to model a variety of micro-architectures. Students studying Electrical and Computer Engineering are most often exposed to microprocessor performance model development and simulation using this tool. Other tools such as Simics[®] [Virutech, 2000], and freeware products such as SimOS [Rosenblum, 1995] and RSIM [Pai, 1997] are also popular.

Although modeling in the areas of engineering that our collective experience covers is very different in many respects, the commonalities encountered in any area of modeling exist also in these areas. The fundamental concepts are the same. The manner in which these concepts are implemented into a model varies. The diversity and commonality of modeling and optimization strategies and methods within chemical, industrial, civil, and electrical engineering can be used to strengthen the overall knowledge and growth in the broader area of information technology.

PROPOSED FRAMEWORK

We propose a framework that enables the integration and subsequent simulation of models from different engineering disciplines to support the simulation of rare events in complex, large-scale systems. This framework will enable the integration of continuous-time, discrete-time, and time-independent (or static) models. It will be implemented in an interfacing language that functions underneath a user interface. By implementing the language in a GUI, the need for a user to learn a new language will be eliminated.

A prototype model of a scaled-down complex system is shown in Figure 1. This complex model contains three sub-models: (1) a discrete-event traffic flow model, (2) a static model of load on a highway bridge, and (3) a continuous-time contaminant dispersion model. Each individual sub-model is defined by its type (e.g. continuous-time, discrete-time, or static), its function (e.g., master or slave model), and its outputs and required inputs. We will use composition diagrams supported by a graphical model editor (similar to that implemented by Modelica). In this graphical editor, the user will position icons that represent the sub-models of the composite model, draw connections between the sub-models to specify the communication directions, and set input/output parameters (data types, variables, values, units), and model type and function in dialogue boxes.

We will base the communication between the different models on the synchronous data flow principle. The sub-model that is defined as the “master” model will drive the simulation. If the master is a discrete-time model, at each (global) time-step this model will halt computation in all continuous-time sub-models, perform its computations, send the results (outputs) to the appropriate sub-models as defined by the communication directions specified in the composite diagram, re-start computation in continuous-time sub-models, then halt its computation. The continuous-time sub-models continue to compute until the next (global) time-step, send any results to the appropriate sub-models, re-start computation in discrete-time sub-models, then halt their computation. Communication between sub-models continues in this manner until simulation completion.

We intend to carefully study the syntax and semantics of existing languages that support combined models such as Modelica, SMILE, and χ since each of these implements some form of communication between sub-models of a composite model.

Prototype Description

In the first stage of the project, a simplified test system was created to simulate the interaction of the various interdependent parts (represented by different computational models) within a single modeling environment for the analysis of a rare event. For this purpose, the collapse of a reinforced-concrete, highway bridge was chosen as the scenario.

The simulation of a bridge structural collapse as the subject of this trial prototype is justified by the fact that about 20 new bridges are added every year to the list of bridges with below-standard performance in the State of New Mexico, mainly due to weakening of structural elements caused by concrete cracking and overloading. In addition, various highways in this region serve as corridors for the transport of industrial and waste contaminant materials, increasing the potential for a regional environmental disaster related to traffic flow.

The continuous traffic flow through the bridge was modeled within ProModel[®] for Windows. Typical traffic and bridge data of an US interstate was used for the simulation, including average daily vehicle and truck data, vehicle type, weight, speed, and nature of the transported materials, bridge dimensions and load capacity, etc. Each event in the traffic model required feedback from physical models (bridge stability and contaminant dispersion models). These physical models determine whether a rare event (bridge collapse) will occur. In case of the bridge collapse and depending on the material transported by the trucks involved in the bridge collapse, the potential spreading of contaminants through air and water to nearby populated areas was assessed. A

schematic of the model, its components and input/output data, is shown in Figure 1.

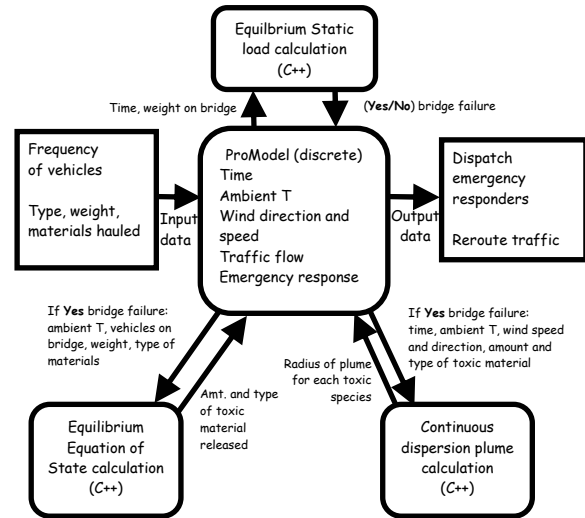


Figure 1: Schematic of hybrid model

The ProModel system outputs data pertaining to the traffic flow that is used by the physical bridge model to determine the stability of the bridge. After the bridge collapse, the amount of contaminant release is determined by a static equation of state model. The amount of contaminants is fed back to the ProModel system, which then communicates with the continuous dispersion plume calculation model to determine spread of the contaminants. The spread of the contaminant information is sent back to the ProModel system, which then reroutes traffic and dispatches emergency responders. The physical models, although computationally simple in this initial prototype, were written in C++ and not using code blocks structured within ProModel, as is usually done. Later stages of the project will consider more complex prototype systems.

CONCLUSIONS

The bridge collapse prototype described in this research is a solid first step in developing a new paradigm for integrating discrete and continuous models in a rare event simulation. The new paradigm will enhance flexibility in modeling and simulation because the constraint of developing all models within a particular modeling environment is removed. This will result faster development of hybrid models because the wealth of existing models, in different environments, can simply be linked simply, rather than reimplemented.

REFERENCES

- "SBETA Program documentation," (1994) Peekel Instruments, Rotterdam.
- Alur, R., Courcoubetis, C., Halbwachs, N., Henzinger, T.A., Ho, P., Nicollin, X., Olivero, A., Skfakis, J., Yovine, S., "The Algorithmic Analysis of Hybrid Systems," (1995) TCS 138 (1): 3-34.
- Azadivar, F. and Tompkins, G., "Simulation optimization with qualitative variables and structural model changes: A genetic algorithm approach," (1999) *European Journal of Operational Research*. Amsterdam; Feb 16, 1999. Vol. 113, Issue 1. pp. 169-192.
- Baine, C., The Fantastical Engineer: A Thrillseekers's Guide to Careers in Theme Park Engineering, (2000) Bonamy Publishing.
- Bandini, P., "Numerical Limit Analysis for Slope Stability and Bearing Capacity Calculations." (2002) Ph.D. Dissertation, Purdue University, West Lafayette, Ind.
- Barros, F.J., "Modeling and Simulation of Dynamic Structure Heterogeneous Flow Systems," (2002) *Simulation* 78 (1).
- Berger, D., Austin, T.M., "The SimpleScalar Tool Set, Version 2.0," (1997) *Technical Report 1342*, University of Wisconsin Computer Sciences Department.
- Blunden, B., and Indraratna, B., "Pyrite Oxidation Model for Assessing Ground-Water Management Strategies in Acid Sulfate Soils." (2001) *Journal of Geotechnical and Geoenvironmental Engineering*, ASCE, 127(2) 146-157.
- Branicky, M.S., "General hybrid dynamical systems: Modeling, analysis, and control," (1996) In R. Alur, T.A. Henzinger, and E.D. Sontag (eds.) *Hybrid Systems III: Verification and Control*, vol. 1066, Lecture Notes in Computer Science, pp. 186-200. Berlin: Springer, 1996.
- Cervenka, V. and Cervenka, J., "Computer Simulation as a Design Tool for Concrete Structures," (1996) *Second International Conference in Civil Engineering*, ICCE-96, Bahrain.
- Choo, L.-P. and Yanful, E. K., "Water Flow Through Cover Soils Using Modeling and Experimental Methods," (2000) *Journal of Geotechnical and Geoenvironmental Engineering*, ASCE, 126(4) 324-334.
- Cook, J., Oliver, R.L., Johnson, E.E., "Toward Reducing Processor Simulation Time by Using Adaptive Modeling Complexity," (2002) *ACM International Conference on Measurement and Modeling of Computer Systems* pp. 252-253.
- Fabian, G., "A Language and Simulator for Hybrid Systems," (1999) Ph.D. Dissertation, Technische Universteit Eindhoven, Budapest, Hungary.
- Fahrland, D.A., "Combined discrete event continuous systems simulation," (1970) *Simulation* 14 (2): 61-72.
- Feyzbakhsh, A. and Matsui, M., "Adam-Eve-like genetic algorithm: a methodology for optimal design of a simple flexible assembly system," (1999) *Computers & Industrial Engineering* Vol. 36 pp. 233-258.
- Flores-Mendoza, C. and Riley, L. "Optimizing the Global Facility Design Problem with Simulation and Soft Computing Methods," (2002) *Proceedings, 2002 Advanced Simulation Technologies Conference (ASTC 2002)*.
- Haddock, J. and Mittenthal, J., "Simulation optimization using simulated annealing," (1992) *Computers & Industrial Engineering*. Vol. 22 Issue 4; pp. 387-395.
- Henzinger, T.A., "The Theory of Hybrid Automata," (1996) Proceedings of the 11th Annual Symposium on Logic in Computer Science (LICS) pp. 278-292.
- Hoare, C.A.R., "Communicating Sequential Processes," (1978) *Communications of the ACM*, pp. 666-677.
- Hu, Y. and Randolph, M. F., "Deep Penetration of Shallow Foundations on Non-homogeneous Soil," (1998) *Soils and Foundations*, JGS, Tokyo, 38(1) 241-246.
- Hu, Y., Randolph, M. F.; and Watson, P. G., "Bearing Response of Skirted Foundation on Nonhomogeneous Soil," (1999) *Journal of Geotechnical and Geoenvironmental Engineering*, ASCE, 125(11) 924-935.

- Khire, M. V., and Benson, C. H., "Capillarity Barriers: Design Variables and Water Balance," (2000) *Journal of Geotechnical and Geoenvironmental Engineering*, ASCE, 126(8) 695-708.
- Kim, J. M., Salgado, R., and Lee, J., "Stability Analysis of Complex Soil Slopes Using Limit Analysis," (2002) *Journal of Geotechnical and Geoenvironmental Engineering*, ASCE, 128(7) 546-557.
- Kim, J. M., Salgado, R., and Yu, H. S., "Limit Analysis of Soil Slopes Subjected to Pore-Water Pressures," (1999) *Journal of Geotechnical and Geoenvironmental Engineering*, ASCE, 125(1) 49-58.
- Klingener, J.F., "Combined Discrete-Continuous Simulation Models in ProModel for Windows," (1995) *Proceedings of the 1995 Winter Simulation Conference(SCS)*.
- Kuo, S.S., Chen, E.J., Selikson, P.L. and Lee, Y.M., "Modeling Continuous Flow with Discrete-Event Simulation," (2001) *Proceedings of the 2001 Winter Simulation Conference (SCS)*.
- Law, A.M. and Kelton, W.D., Simulation Modeling and Analysis, (2000) 3rd Edition, New York, NY: McGraw Hill.
- Lee, S., Pritchett, A., Goldsman, D., "Hybrid Agent-Based Simulation for Analyzing the National Airspace System," (2001) *Proceedings of the 2001 Winter Simulation Conference (SCS)*.
- Lei, H., Francis, L.F, Gerberich, W.W., & Scriven, L.E., "Stress development in drying coatings after solidification," (2002) *AIChE Journal*, **48**:437.
- Lipchin, A. & Brown, R.A. "Hybrid finite-volume/finite-element simulation of heat transfer and melt turbulence in Czochralski crystal growth of silicon," (2000) *Journal of Crystal Growth*, **216**:192.
- Lisitsyn, N.V., Krivospitskii, A.N., & Kuzichkin, N.V., "Optimum control of a crude oil processing plant," (2002) *Theoretical Foundations of Chemical Engineering*, **36**:273.
- Liu, J., "The impact of neighborhood size on the process of simulated annealing: Computational experiments on the flowshop scheduling problem," (1999) *Computers & Industrial Engineering*; New York; Vol. 27 Issue 1,2; pp. 285-288.
- Loukidis, D., Bandini, P., and Salgado, R., "Stability of seismically loaded slopes using limit analysis," (2003) *Géotechnique*, 53(5), 463-479.
- Luna, R. and Wu, Yong, "Simulation of Temperature and Stress Fields During RCC Dam Construction," (2002) *Journal of Geotechnical and Geoenvironmental Engineering*, ASCE, 126(5) 381-388.
- Mattsson, S.E. and Otter, M. and Elmqvist, H. "Modelica Hybrid Modeling and Efficient Simulation," (1999) *Proceedings of the 38th IEEE Conference on Decision Control (CDC)*.
- Mitchell, M.C., Fort, A. K., Williams, L., Wilson, D.B., & Saulsberry, R., "Thermodynamic Analysis of Compression Heating of MIL-SPEC Unsymmetric Dimethylhydrazine," (2002a) *Proceedings of the 51st JANNAF Propulsion Meeting*.
- Mitchell, M.C., Hendren, Z., Brown, M., Wilson, D.B., & Saulsberry, R., "Thermodynamic Analysis of Compression Heating of Two-Phase Monomethylhydrazine," (2002b) *Proceedings of the 51st JANNAF Propulsion Meeting*.
- Music, G., and Matko, D., "Combined simulation for process control: extension of a general purpose simulation tool," (1999) *Computers in Industry*, 38 (2) 79-92.
- Pai, V.S., Ranganathan, P., Adve, S.V., "RSIM: An Execution-Driven Simulator for ILP-Based Shared-Memory Multiprocessors and Uniprocessors," (1997) *Workshop on Computer Education (HPCA)*.
- Pierreval, H. and Tautou, L., "Using evolutionary algorithms and simulation for the optimization of manufacturing systems," (1996) French Institute of Mechanical Engineering (IFMA) Aubiere Cedex France.
- Pritsker, A., Sigal, C.E., and Hammesfahr, R.D., SLAMII network models for decision support, (1989) Englewood Cliffs, NJ: Prentice-Hall.
- Riley, L. and Flores-Mendoza, C., "A New Global Optimization Approach for Solving the Integrated Facility Design Problem," (2001) *Proceedings, 2001 Advanced Simulation Technologies Conference (SCS - ASTC 2001)*.

- Riley, L. and Flores-Mendoza, C., "Using Static and Dynamic Simulation to Create a New Global Optimization Method for Solving the Integrated Facility Design Problem," (2001) *Proceedings, 2001 Summer Simulation Conference (SCS - ASTC 2001)*.
- Riley, L. and Gomez-Sanchez, M., "Integrating Dynamic Simulation and Genetic Optimization Methods for Solving Stochastic Network Problems," (2001) *Proceedings, 2001 Advanced Simulation Technologies Conference (SCS - ASTC 2001)*.
- Riley, L. and Gomez-Sanchez, M., "Networks and Simulation Modeling for Solving Stochastic Routing Problems," (2001) *Proceedings, 2001 Summer Simulation Conference (SCS - ASTC 2001)*.
- Riley, L. and Salais, T., "Neural Genetic Approach to the Stochastic Scheduling Problem with Set-Up Times," (2001) *Proceedings, 2001 Advanced Simulation Technologies Conference (SCS - ASTC 2001)*.
- Riley, L. and Salais, T., "Using Simulation to Design, Implement and Validate a Fuzzy Neural Model to Solve Job Shop Scheduling Problems," (2001) *Proceedings, 2001 Summer Simulation Conference (SCS - ASTC 2001)*.
- Riley, L., "Methodological Perspectives for optimizing discrete-event simulation models," (2000a) *Proceedings, 2000 Advanced Simulation Technologies Conference (SCS - ASTC 2000)*.
- Riley, L., "The Impact of Technology on Industrial Engineering," "Impacto de la Tecnología en la Ingeniería Industrial," (2001) *Proceedings of the 1st International Symposium on Technology in Engineering and the Sciences*, Juarez, Mexico.
- Riley, L., "Using Optimized Simulation As A Tool To Create And Support Intelligent Manufacturing Systems," (2000b) *Proceedings, The Third World Congress on Intelligent Manufacturing Processes & Systems*, MIT, Cambridge, Massachusetts.
- Rosenblum, M., Herrod, S.A., Witchel, E., Gupta, A., "Complete Computer System Simulation: The SimOS Approach," *IEEE Parallel and Distributed Technology*.
- Salais, T., and Riley, L., "Neuro-Genetic Stochastic Scheduling Problems," (2002) *Proceedings, 2002 Advanced Simulation Technologies Conference (ASTC 2002)*.
- Schriber, T. J., "Introduction to GPSS," (1986) *Proceedings of the 1986 Winter Simulation Conference*, 75-86.
- Shi, L., Olafsson, S. and Chen Q., "A new hybrid optimization algorithm," (1999) *Computers & Industrial Engineering*. Vol. 36; pp. 409-426.
- Smejkal, Q., & Soos, M., "Comparison of computer simulation of reactive distillation using ASPEN PLUS and HYSYS software," (2002) *Chemical Engineering and Processing*, **41**:413.
- Somers, S.A., Ayappa, K.G., McCormick, A.V., & Davis, H.T., "Binary fluids in planar nanopores: Adsorptive selectivity, heat capacity and self-diffusivity," (1996) *Adsorption*, **2**:33.
- Sozen, M. A., Kilic, S. A. and Hoffmann, C. M., "September 11 Pentagon Attack Simulations Using LS-Dyna," (2002). <http://www.cs.purdue.edu/homes/cmh/simulation/>.
- Sturgal, J.R., and Prisbrey, K.A., "Combined Discrete Event and Transfer Function Simulation of Mineral Processing Flowsheets," (2001) *Proceedings of the 3rd International Conference on Intelligent Processing and Manufacturing of Materials*.
- Vartak, B., Kwon, Y.I. Yeckel, A, and Derby, J.J., "An analysis of flow and mass transfer during the solution growth of potassium titanyl phosphate," (2000) *Journal of Crystal Growth*, **210**:704.
- Virutech, "Simics 1.0," (2000) <http://www.simics.com>
- Yu, H.-S., Salgado, R., Sloan, S. W. and Kim, J. M., "Limit Analysis Versus Limit Equilibrium for Slope Stability," (1998) *Journal of Geotechnical and Environmental Engineering*, ASCE, 124(1) 1-11.
- Zeigler, B.P, Praehofer, H., Kim, T.G., Theory of Modeling and Simulation – Integrating Discrete Event and Continuous Complex Dynamic Systems, (2000) 2nd Ed., Academic Press.
- Zolfaghari, S. and Liang, M., "Machine Cell/Part Family Formation Considering Processing Times and Machine Capacities: A Simulated Annealing Approach," (1998) *Computers & Industrial Engineering* Vol. 34 No. 4; pp 813-823.